RADC-TR-88-120
Final Technical Report
May 1988

APPLICATION OF STATISTICAL METHODS TO THE DEVELOPMENT OF NAVAL SOFTWARE MAINTENANCE AND RELATED COST ESTIMATION MODELS

DATA BASE Services

Dr. R. Gulezian

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss AFB, NY 13441-5700

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS N/A
2a. SECURITY CLASSIFICATION AUTHORITY N/A	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) N/A	5. MONITORING ORGANIZATION REPORT NUMBER(S) RADC-TR-88-120
6a. NAME OF PERFORMING ORGANIZATION DATA BASE Services 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Rome Air Development Center (COEE)
6c. ADDRESS (City, State, and ZIP Code) 37 E. Montgomery Ave Ardmore PA 19003	7b. ADDRESS (City, State, and ZIP Code) Griffiss AFB NY 13441-5700
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Naval Underwater Systems Center 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F30602-81-C-0193
8c. ADDRESS (City, State, and ZIP Code) New London Laboratory Code 3344/B1dg 28 New London CT 06320-5594	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO: N/A; PROJECT NO: N123; TASK NO: 00; WORK UNIT ACCESSION NO: P1
11. TITLE (Include Security Classification) APPLICATION OF STATISTICAL METHODS TO THE DEVELOPMENT OF NAVAL SOFTWARE MAINTENANCE AND RELATED COST ESTIMATION MODELS	
12. PERSONAL AUTHOR(S) Dr. R. Gulezian	
13a. TYPE OF REPORT Final	13b. TIME COVERED FROM Aug 87 TO Apr 88
14. DATE OF REPORT (Year, Month, Day) May 1988	15. PAGE COUNT 60
16. SUPPLEMENTARY NOTATION N/A	

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	
12	05		Software Cost Models
			Software Maintenance
			Statistics

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report describes a framework within which the Sonar Systems Department of the Naval Underwater Systems Center would be able to develop a model over time that can be used to effectively estimate software maintenance costs. The primary focus of this study is on the development of a framework by which statistical methods can meaningfully contribute to the software cost modeling process and model structures. Although the report describes a framework in the context of sonar systems software maintenance, the framework is generalizable to the future structure of all software cost models and model building processes.

AD/SPO-TR-88-120 (FR)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Andrew J. Chruscicki	22b. TELEPHONE (Include Area Code) (315) 330-4063 22c. OFFICE SYMBOL RADC (COEE)

DD Form 1473, JUN 86 *Previous editions are obsolete.*

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

PREFACE

The need to estimate the cost of developing and maintaining military software is unquestionable. To date, the results of much accumulated effort and activity are available with respect to cost analysis, software engineering, metric development, modeling and data collection. Due to the immensity and importance of the problem and an on-going need for more realistic and less error-prone estimates, another generation of more sophisticated effort is emerging which is directed toward unifying the results of individualized efforts to date and a search for a methodology that will meet the on-going and ever-growing need. In part, this requires the use of statistical methods, which slowly is being recognized within the industry.

The current effort was initiated in order to establish a beginning framework within which the Sonar Systems Department of the Naval Underwater Systems Center, New London would be able to develop a model over time that can be used to effectively estimate software costs. Based upon recognition within the Department regarding the overall problem and the role played by statistics, a secondary objective automatically satisfied by this effort is the development of a structure by which statistical methods can meaningfully contribute to the cost modeling process.

ACKNOWLEDGEMENT

The author would like to give special thanks to Andrew Chruscicki, Rome Air Development Center, for his continued support and clarification of many of the ideas presented and for his careful review of various drafts of this report. Also, the effort could not have been accomplished without the initial support and continual contributions made by Ahlam Shalhout, Naval Underwater Systems Center. Finally, thanks to Captain Joseph Dean, AF Electronics Surveillance Division, for his initial recognition of the underlying ideas and concepts presented, and for his helpful review of the last draft of the report.

CONTENTS

INTRODUCTORY REMARKS

The purpose of this report is to present the results of an effort to establish an initial framework that is directed toward the future development of a statistically based model that can be used to effectively estimate the cost of software maintenance within The Sonar Systems Department of the Naval Underwater Systems Center, New London. This stems from the fact that no environment specific model exists within Sonar Systems to date. The results are based, at the outset, upon an approach that can combine the useful features associated with the existing state of the art embodied within the publicly available models, and methods of statistics and modeling heretofore not utilized in software cost modeling in their entirety.

Not only does the report consider requirements and activities necessary to develop an on-going modeling effort that applies to software maintainance cost estimation, but it considers a structure within which other software estimation needs within the department can be addressed within the same structure. In addition, the report provides the necessary backdrop relating to the existing state of software cost modeling as it pertains to the unique aspects of the approach, which draws heavily upon the use of statistical methods. Results of the effort and the recommended modeling structure that subsequently can be developed from it can realistically be applied to new environments to determine appropriate models and the potential importance of the variables relevant to cost.

The following topics are covered in sequence in the report:

Sonar Systems Department

Sonar Systems Estimation Needs

General Framework

Existing Models

Calibration and Reformulation

Recommended Model Features

Statistical Modeling

Basic Requirements

Interface

Support Requirements

Concluding Remarks

Appearing at the end of the report are a list of references.

It should be noted at the outset that the report is directed toward providing information regarding potential modeling approaches and the ability to select priorities within a defined framework. It is not, therefore, intended to provide actual decisions or choices, this being within the purview of individual members of the Sonar Systems Department.

SONAR SYSTEMS

The Sonar Systems Department deals primarily with the development and maintenance of five subsystems which comprise the surface ships sensor suite associated with the AN/SQQ-89V system. The five subsystems are given as follows:

AN/SQQ-28. Sonar Signal Processing System

AN/SQS-53B. Hull Mounted Sonar

AN/SQR-19. Tactical Towed Array Sonar System

AN/UYQ-25. Data Processing System

MK/116. Fire Control

Actually, the MK-116 is under development at the Naval Ocean Systems Center (NOSC), San Diego, but represents part of the entire system. SQQ-28 and UYQ-25 have been undergoing maintenance for periods of three and seven years, respectively. Maintenance is characterized mainly in terms of fault repair and upgrades to the existing subsystems. Fault repair is initiated via a problem trouble report (PTR), which when implemented is included within an engineering change proposal (ECP). Multiple faults may be embodied within a single ECP. Upgrades also are implemented through an ECP.

Internally, the Department maintains a data base for configuration control which is potentially expandable to include additional data required for modeling purposes. An additional pre-packaged configuration control program is being considered for purchase within the next year. Although not clearly established at this time, it appears that any data needed for modeling purposes must be incorporated within configuration control. Consequently, for those needs not corresponding to projects under configuration control, special attention must be given in the future.

SONAR SYSTEMS ESTIMATION NEEDS

Based upon discussions with personnel within the Sonar Systems Department, the following seven types of estimates that are needed with respect to the above subsystems now or in the future were established:

(1) Annual software maintenance budget

(2) 5-year annual moving software maintenance budget

(3) Cost to correct a fault during O & M

(4) Cost to implement an upgrade during operations and maintenance

(5) Development cost prior to the operation and maintenance phase

(6) Cost of R & D prototype development

(7) Cost to implement an ECP during development

Although itemized separately, activities relating to the documented needs are not mutually exclusive. On the one hand, the annual maintenance cost is comprised of the sum of the individual costs of fault repairs and upgrades. On the other hand separate phases designated in actual development are employed in the maintenance activity. The key difference in activity lies in the manner in which fault repairs and upgrades are formally tested and integrated into each subsystem. Also, in the case of maintenance, an entire configuration item may be compiled rather than smaller individual components containing the modification, which is true in actual development. In many cases, due to their accessibility, development models actually are used to estimate aggregate maintenance costs.

The estimation needs are broken down as they are for several reasons. Potentially, different ways of approaching the problem and different types of data and data sources can be applied to the individual cases. Moreover, various needs emanate from different administrative levels and have different priorities associated with them. Due to the interrelated nature of the types of estimates required, and the existing state of the art, all will be considered to varying degrees.

When viewing the cost modeling problem over the long-term, each of the estimating needs eventually can and should be embraced within the same model, although different model components may apply to different cases. Here, much depends upon the data available at particular points in time, which represents one of the biggest hurtles to overcome. The approach taken in this report

attempts to deal with the problem as comprehensively as possible, both in terms of types of estimation needs and varying degrees of data availability.

In addition to the overlap among activities relating to the estimation needs, relationships among the individual estimation problems exist for which some clarification is helpful in terms of the ideas that follow. One of the goals, estimating the annual maintenance budget, can be dealt with basically in one of three ways: (1) an aggregate figure expressed as a portion of incurred or estimated development effort or cost, (2) an aggregate figure based upon a historical record of annual maintenance, available capacity and maintenance planning, or (3) in terms of the accumulated estimates of anticipated or expected fault repairs and upgrades.

The 5-year moving maintenance budget could be considered similarly or it also could account for the phases not completed in earlier maintenance efforts. In the case of the disaggregated annual estimates, additional estimates of the number of faults and upgrades not anticipated or generated would have to be employed.

Individual estimates comprising the total estimated annual maintenance budget would be obtained similarly to meet the third and fourth estimation needs specified above. Cost to implement an ECP during actual development can be treated as one during operations and maintenance with consideration given to the phase of development in which it is introduced. Actual development and prototype development cost from a modeling point of view can be dealt with in a similar manner, recognizing of course that prototypes would be smaller in size and to date are not under configuration control.

The discussion presented in the previous paragraphs has been introduced to indicate briefly the overlap among the various estimating needs as they relate to potential model development. Basically, this is important since parts of these needs can be accommodated by the same model structure and

approach. It should be recognized that the term model structure is to be distinguished from a specific model. Different specific models may possess different numerical valued coefficients and different functional forms yet fall within the same structure. Ideally, although currently not realizeable due to practical limitations, all software cost estimation problems could be embraced by a single model structure.

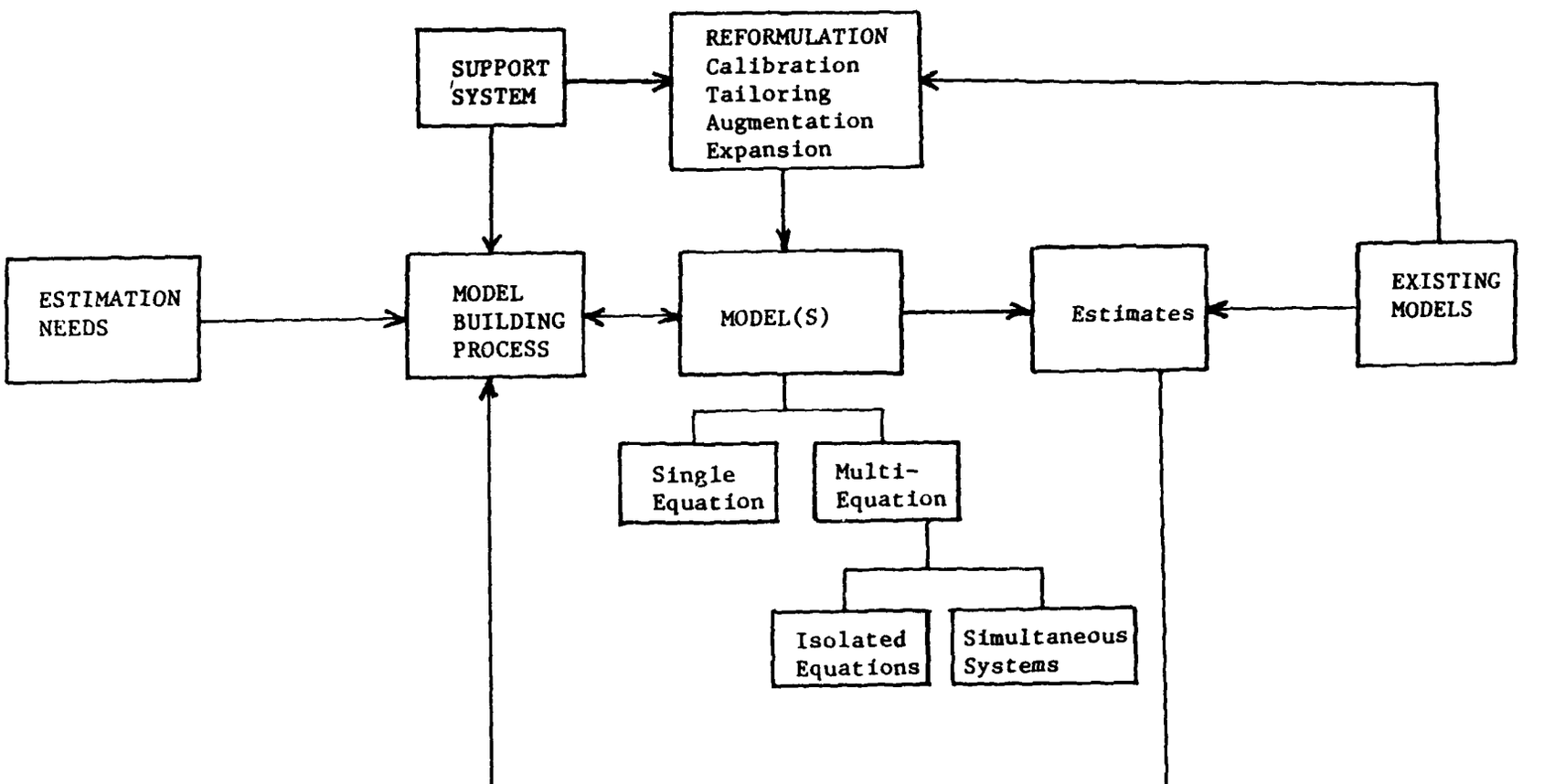
GENERAL FRAMEWORK

The purpose of this section is to provide a framework that addresses modeling and estimating alternatives using ideas that form a basis for introducing and unifying the individual items presented later. A brief description of the framework is depicted in Figure 1. Parts of the framework as presented are unique relative to traditional approaches to software cost modeling. The two most unique features presented in this report are the contribution made by statistical methods and the manner in which these methods can be used to bridge the gap between the current state of the art and needed future modeling efforts.

Briefly, it can be seen from the figure that a modeled estimate, as opposed to one, say, made by analogy, judgment, or intuition, can be generated from one of four sources: (1) an existing or publicly available model, (2) a potentially modified version of a publicly available model, (3) an environment or system-specific model internal to the facility requiring the estimate, or (4) a combination of the previous three.

Figure 1

GENERAL MODELING STRUCTURE



Since none of the existing or publicly available models truly satisfy Sonar Systems' estimation needs, the two reasonable courses of action to be considered are modification of a publicly available model or development of an environment or system-specific model. Actually, sufficient data, if any, is not available in order to either properly calibrate and/or validate any of the existing models in order to determine the extent to which the models are applicable to any of the stated needs. A full environment specific data base really should be used in order to adequately validate the models with respect to estimated output. The same is true, assuming certain conditions were to be met, in order to properly calibrate the models.

Depending upon the particular model certain amounts of information, however, can be utilized which reflects the level of knowledge available, or the state of the art to date. These can be used as starting and/or reference points for a new modeling effort or can, potentially, be used to make estimates. Such estimates should, of course, be incorporated into a structured validation process.

A start from scratch modeling effort might be the most appropriate route to undertake, however, the inital start-up definitional and data definition and collection effort is somewhat time-consuming and rather prohibitive. Although various general modeling approaches are available, the approach to be suggested here, which is primarily statistical in nature, requires certain additional considerations heretofore not used in software cost estimation modeling. As will be discussed, such procedures can incorporate the individual contributions from the existing models that allows for a more timely and more reasonable starting position than by using existing models alone or by starting from scratch.

Models as we are considering them here can be viewed as single equation or multi-equation models. A single equation model can be considered as one

such that a single variable, or parameter, to be predicted is represented as a function of one or more independent, or explanatory, variables. For example, in the case where cost and schedule length are to be estimated and done so by two separate equations, one would be dealing with two single equation models.

If, on the other hand, some or all of the independent variables, themselves, are represented as a function of variables in the system in the form of added equations, such a set of equations can be thought of as a multi-equation model. For example, if schedule length is represented by a separate equation but also is used as one of the variables to predict cost or effort, this can be considered to be a two equation model. As another case in point, if, for example, separate equations are used to obtain effort estimates at the system, CSCI, CSC, and module levels, one would be dealing with a multi-equation effort model. In point of fact, many of the publicly available software cost models are of this latter form.

A key point to be made with regard to the above comments is that based upon the typically used-heuristic methods of model development employed currently, the individual equations are treated in isolation. Although this is an acceptable way in which to treat the problem based upon the methods employed, it is also possible to consider the variables and equations in the system statistically as a simultaneous system, accounting for all of the individual interrelations and error conditions that exist, [5]. Such treatment would lead to dynamic models, but a much more complex modeling process. This ultimately can be considered as the state of the art as software engineering advances. Although more will be said toward the end about multi-equation modeling, the emphasis in the report is in terms of isolated single equation models.

Before considering the primary thrust of the figure, it is beneficial at this point to define the terms used in the box starting with the term

reformulation. Here, the term reformulation is used in conjunction with an already developed publicly available model which more or less has been heuristically developed. A reformulated model would be one that has the same equation structure as the original but for which all multipliers or coefficients are simulataneously estimated applying statistical regression procedures using the data base upon which the original model was developed. If successful, a reformulation as defined would not possess coefficient values that are identical to the original model, but should possess the same degree of fit as the original as well as similar error characteristics and predictive ability.

Typically, the term calibration is used to mean that new coefficients or multipliers for an existing model are established or modified such that the same model structure applies to a data base or an individual system corresponding to an environment other than the one upon which the model was developed. Various methods are used to calibrate a model. For purposes of this report, it is assumed that a true calibration must be applied to an entire data base rather than an individual observation. Further, here we assume that a calibrated model is one that is a successfully reformulated model in the sense used above, but with respect to an entire data base corresponding to an environment other than the one upon which the original model was developed.

Although additional terms which are important apply to the modeling process in general, these are introduced here since they also relate to the concepts with respect to existing models which are presented next. The term tailoring refers to the modification of a model by adding nonlinear terms of existing variables, deleting variables, introducing structural changes to the model, or deleting or reweighting individual observations with the goal of better predictability.

By augmenting a model we shall mean that additional observational units, such as added CSCI's, CSC's, or modules have been included upon which all coefficients or multipliers have been re-estimated based upon the entire newly augmented data set. Expansion of a model shall mean that additional factors are considered in the model in the form of new measures or metrics, interrelations among drivers, dependencies and constraints, or added equations.

In each case in which a model is modified, all coefficients or multipliers, new and old, would be re-estimated. Formerly introduced coefficient values will be numerically different, however, whether the differences are of importance or not must be established. Part of this process is executed with the use of statistical methods. Once a modified model has been tested or accepted to be superior to the previous one, we shall refer to it as being reconfigured.

The four sources of a modeled estimate, or types of models, namely existing models, a start from scratch model, a reconfigured existing model, or a combination of these can be used individually as a basis for making estimates within a particular facility. Each underlying approach, has its own set of advantages or limitations, a time frame within which estimates are available, and a different type and amount of effort in order to obtain a useful estimate. Although the choice must be made internally regarding the approach(es) to be used, the chart depicted in Figure 1 suggests that useful components from all sources can be utilized. Moreover, information from all sources can be utilized as part of a loop that provides reconfigured models on an on-going basis. Details regarding the contribution of statistical methods and the required support system required to implement such an approach appear in subsequent sections of the report.

EXISTING MODELS

Numerous models are in existence for estimating the cost of software, of which some are developed and used exclusively within various organizations while others are publicly available in one form or another through purchase or lease, or they appear in the published literature. Many of these models are described, to varying degrees, in Bailey, et. al. [1] and Boehm [3]. Of these, four particular models are notable due to their wide-spread use and acceptance and are to be considered here. The four models are COCOMO [3], SLIM [13], System-3 [11], and Price-S [12]. Due to the amount of information available about COCOMO, many versions and variations of the model are used privately or are available publicly. With respect to COCOMO, only the original form cited in Boehm [3] will be considered here.

There are a number of reasons for introducing the four models at this point. On the one hand, all of these models have deficiencies in terms of their appliciability to new environments beyond those used to develop the models. These deficiencies, therefore, form a partial basis for establishing requirements necessary to develop environment-specific models. Also, despite these deficiencies, the models serve as a reference point for new modeling efforts and supply certain fundamental inputs as a starting point to these efforts. On the other hand, parts of the material presented is necessary in order to introduce reformulated versions of the models in order to establish a proper interface with new modeling efforts, especially those based upon use of more advanced statistical methods.

The discussion of the four models below does not provide a complete description of each. More complete descriptions can be found in the references cited above. The intent here is to highlight those features that

bear directly upon this effort as it relates to Sonar Systems' estimation needs.

In general, the models considered are structured as one or more equations or equivalent procedures that can be reconstructed in equation form, or as a cost estimating relationship, or CER. Each relates effort as the dependent variable to varying numbers of cost driver variables; the Price model employs cost directly as the dependent variable, however, this is being changed in order to conform to the practice used by the other models, since effort is independent of inflationary factors over time. The key points to be discussed are model form and structure, nature of the inputs to obtain an estimate, data bases employed and corresponding representation, generalizeability of results, statistical features, and calibration.

The degree to which knowledge regarding the specific form of each of the models varies. Only COCOMO is a complete white box model, which, means that not only is the concomitant equation form known, but so are the numerical values of the coefficients, or multipliers. In other words, the exact way in which the inputs are processed in order to obtain a cost or effort estimate is explicitly known. The Price model, on the other hand, is a complete block box model, since neither the equation form nor the individual coefficient values are known.

The remaining two models, SLIM and System-3, presumeably are partial white boxes since the basic equation structure for each is published. It appears, however, that additional adjustment factors or multipliers have been incorporated into each model such that the way these are used do not appear in the published material.

Based upon information that is available, all of the models can be classed as multiplicative in the sense that each can be linearized in logarithmic form. SLIM and System-3 are unique to the extent that they

incorporate a Rayleigh curve, [3], which is exponential, into the model structure. Since an exponential form can be linearized through a logarithmic transformation, it will be classed as a multiplicative form also. This becomes important when presenting considerations relating to the potential contributions of each of the models to the development of a model(s) for Sonar Systems.

The original data base upon which COCOMO was developed is published in its entirety in Boehm, [3]. This corresponds to both the Basic and Intermediate forms of the model which deal with the system level exclusively. Data for the Detailed version, which considers the subsystem and module levels, have not been published and apparently are no longer available within TRW. No specific data upon which each of the other models was developed is publicly accessible, and in the case of System-3 the data, apparently, are no longer available to the developer or the current vendor of the model. Accessibility to the underlying data base is important in order to assess the representation of systems considered by a particular model. Moreover, the original data can be used to develop a statistically reformulated version of the models that serves as a starting point for environment-specific model development with additional features not possessed by the original models, and which, can be formulated prior to the long-term data collection effort required to develop a truly environment specific model.

The problem of adequate representation relates partially to the extent to which models can be generalized to other environments. Proper validation procedures must be employed in order to determine the extent to which models developed on the basis of one set of systems can be applied to other systems, especially military systems. Of greater importance with regard to the problem of generalizeability is the manner in which the models were developed and are applied. Although objective curve fitting techniques have been used to obtain

components of the models, heuristic methods to varying degrees have been employed which tend to impose a solution or "good fit" to the data employed. This neither allows the data to determine the nature of the model nor does it lend itself to any appreciable generalization of the resulting coefficients or the resulting estimates. Moreover, a good fit to an existing data set alone does not guarantee good predictions. Although the nature of the interrelationships, that by necessity must exist, among the cost drivers would be considered in the process, no objective way of dealing with the interrelationships in terms of variable selection and contribution is attached to the methodology.

Except for sizing and constraint variables, much of the input data used in the various models typically is in the form of subjective ratings corresponding to various cost driver attributes. Actual quantities and the values used for each of the models can be found in Bailey [1]. A simpler source presenting these quantities for all four models is Najberg [8]. Although more meaningful measured or metric data presumeably could be used to expand the models as it is accumulated, assuming the model is a complete white box, no systematically objective way of doing this exists using the current procedures underlying the publicly available models. Furthermore, due the manner in which the models were developed in conjunction with the qualitatively rated inputs precludes extrapolation or interpolation based upon values not predesignated within the rating scales.

The only so-called statistical feature possessed by any of the models is the construction of a probabilistic error bound or interval on the effort estimate. This feature is included in SLIM, Price, and System-3. Basically, this is accomplished by means of a simulation, presumeably assuming an underlying normal distribuiton with some pre-designated numerical value for the standard deviation as an input. This procedure is acceptable up to a

point, however, it does not properly account for the nature of prediction error, which can be analytically based using other modeling methods. Moreover, properly measured error can be used in many ways in the modeling process to develop more effective models. This is not considered whatsoever with respect to the existing models.

Calibration and Reformulation

In order for any of the existing models to be useful in an environment other than the one in which it was developed, it should be calibrated to the new environment in the sense defined above. In the case of COCOMO, a procedure is suggested which is based upon an entirely new data set to do this, but only with respect to the two nominal coefficients initially relating effort to lines of code. Although not suggested, the only means for calibrating the remaining cost driver multipliers is by trial and error procedures. Aside from the fact that such procedures have no potential generalizeability, the values of the multipliers chosen are restricted to the values predesignated within the model structure and cannot assume any other values.

The suggested procedure for calibrating the other models is based upon modifying one or two designated multipliers until the modeled estimate equals some predesignated value, either one desired or a past actual corresponding to a potentially similar system. Since the numerical values of the coefficients in the remaining models are not provided, calibration to a complete data set reduces to manipulating the inputs on a trial and error basis until a "proper fit" is achieved. Although limited in scope, only COCOMO can be fully calibrated in terms of all of its multipliers. In order to be properly calibrated, the model should be statistically reformulated.

Since the complete structure of the COCOMO model and the complete original data base is known, the COCOMO model can be reformulated as defined, Gulezian [5]. This is not the case for the other models. The Intermediate COCOMO model, which is the one appropriate for the present needs, can be written in equation form as

$$MMest = a(KDSI)^b \ (EAF) \tag{1}$$

$$= a(KDSI)^b \prod_{\substack{i=1 \\ j=1,\,2,\ldots,\,k_i}}^{15} C_{ij}, \tag{2}$$

for which the following symbols are defined:

MM = mammonths or effort

est = estimated

EAF = effort adjustment factor

KDSI = thousand lines of delivered source code (adjusted for or net of re-useable code)

a,b = numerical coefficients corresponding to each of three application modes that initially relate effort to lines of code

C_{ij} = cost driver multiplier corresponding to the j-th category selected for i-th cost driver

k_i = number of rating categories corresponding to the i-th cost driver

$\prod$ = product operator

It can easily be seen by examining the alternate forms, (1) and (2), that the effort adjustment factor, EAF, equals the product of the cost driver

multipliers. Each of these are used to proportionately increase or decrease the basic or nominal estimate, $a(KDSI)^b$, depending upon the selected rating corresponding to each cost driver. Separate values of the nominal coefficients, a and b, are supplied for each of the three application modes. Consequently, one can think of (1) or (2) as representing three equations, one for each mode. As presented, the above equations apply to the overall project or system level, however, the same structure can be applied to the subsystem level. Although separate multipliers have been provided, no subsystem data is available.

The Intermediate COCOMO model can be reformulated in a number of ways without altering its structure. One possibility is to rewrite it in terms of indicator or dummy variables as follows:

$$MM_{est} = a(KDSI)^b \prod_{i=1}^{15} \left[\prod_{j=1}^{k_i} C_{ij}^{y_{ij}} \right], \tag{3}$$

$$y_{ij} = 0, 1$$
$$j = 1, 2, \ldots, k_1$$

where the newly introduced term y_{ij} represents a dummy variable which equals 1 when the cost driver multiplier corresponding to the j-th rating category for the i-th cost driver is present, and is zero otherwise. When applied to make an estimate, equation forms (2) and (3) are equivalent since the values of the dummy variables assigned a zero wash out of equation (3).

In order to implement the equation in (3) it is necessary to use the values provided for a, b, and the C_{ij} provided within the COCOMO structure. A single manmouth estimate, therefore, requires seventy-five potential numerical inputs, which are considered as coefficients in (3), for which seventeen are chosen in order to produce an estimate. The inputs, however, are based upon the data base upon which the model was developed.

If, on the other hand, it is of interest to apply the model to a particular system or development environment not related to the COCOMO data base, it is necessary to tailor or calibrate all of the multipliers or coefficients. This implies that the seventy-five coefficients in (3) must be estimated based upon data, either quantitative or qualitative, that is related to the system or development environment for which an estimate is to be made. When simultaneously estimating coefficients in equations similar to (3), the simplest procedure is to linearize the equation by taking logarithms and then apply the method of least squares. In the form presented, however, this is not possible since inclusion of dummy variables for all rating categories corresponding to each cost driver creates linear dependencies which render a solution impossible.

An alternative model for which the coefficients are simulataeously estimable is as follows:

$$MM_{est} = a'(KDSI)^{b'} \prod_{i=1}^{15} \left[\prod_{j=1}^{k_i - 1} C'^{y_{ij}}_{ij} \right], \tag{4}$$

$$y_{ij} = 0, 1$$
$$j = 1, 2, \ldots, k_i$$

where the prime (′) associated with a, b, and the C_{ij} designates an estimable coefficient. The essential difference between (3) and (4) is that one of the rating categories corresponding to each cost driver has been omitted from the equation, resulting in sixty coefficients to be estimated. In light of the fact that the published COCOMO data base contains sixty-three data items or cases, the estimation procedure applied to (4) would not be parsimonious. The term parsimony here is used in a statistical sense, meaning that the number of coefficients to be estimated is 'small'. This contrasts with the definition used in Boehm [1], where parsimonious is used synonomouly with the term non-

redundant. Simply stated, parsimony is important to achieve in order to obtain generalizeable model forms corresponding to which appropriate measures of error can be estimated. Alternatively, if calibration is the main objective, it is desirable to reduce the number of coefficients to be estimated to the smallest possible number since the number of systems or projects corresponding to individual environments typically is sparse.

In order to achieve the basic goals initially presented, which include simultaneous estimability and parsimony, the following model structure was selected:

$$MM_{est} = a'(KDSI)^{b'} \prod_{i=1}^{15} d_i'^{C_{ij}} , \qquad (5)$$

$$j = 1, 2, \ldots, k_i$$

where all of the terms except d_i' have been defined previously. The d_i' represent coefficients that correspond to the fifteen cost drivers that are to be estimated simultaneouly along with the nominal coefficients, a' and b', resulting in twenty-one coefficients to be estimated. With respect to this formulation, the 'data inputs' represent lines of code (KDSI), manmonths (MM) and the C_{ij}, which are the cost driver multipliers corresponing to the j-th rating category associated with the i-th cost driver. Although not intended to be variate values, it can be presumed that the multipliers, which are at worst ordinal in nature, can be viewed as indexes of 'intensity' with some underlying continuity associated with them. Consequently, once the d_i' - values are estimated, a calibrated multiplier can be obtained as

$$\text{Calibrated Multiplier} = d_i'^{C_{ij}} \qquad (6)$$

The values of a′ and b′ resulting from the estimation process automatically would be in calibrated form.

Actually, neither of the equation forms, (4) or (5), contain a provision for distinguishing among the application modes directly within the equations. This can be accomplished easily with additional dummy variables. In order to avoid the problem of linear dependence cited above, the general rule to follow is to employ one less dummy variable than the number of categories corresponing to the categorical or qualitative variable included in the equation.

Since there are three application modes in the COCOMO model, two dummy variables would be required for a complete description. For simplicity, in order to illustrate the suggested methodology, here we shall employ a single dummy variable which distinguishes between the embedded mode and the organic and semi-detached modes combined. The resulting equation takes the form

$$MM_{est} = a'(KDSI)^{b'} (c'^{m}) \prod_{i=1}^{15} d_i'^{C_{ij}} , \qquad (7)$$

$$m = 0, 1$$
$$j = 1, \ldots, k_i$$

The term c′ represents an additional coefficient to be estimated which corresponds to the mode variable, m. This assumes a value of 1 in the case of the embedded mode and a value of zero for either the organic or semi-detached mode. It is this equation, (7), that we shall use as the reformulated model form.

The basic method of solution employed in order to determine the reformulated model is multiple linear regression analysis using traditional least squares. The equation form in (7) easily can be linearized by taking logarithms, which yields

$$\ln(MM_{est}) = A' + b'\ln(KDSI) + C'm + \sum_{\substack{i=1 \\ j = 1, 2, \ldots, k_i}}^{15} D_i' C_{ij} , \qquad (9)$$

where

$$A' = \ln a'$$

$$C' = \ln c'$$

$$D_i' = \ln d_i', \; i = 1, \ldots, 15$$

and ln denotes the natural logarthm.

The procedure, therefore, is to estimate A', C', b', and the D_i' directly by minimizing the sum of squares of the logarithm of manmonths and then converting the results into orginal units. Substituting the specific numerical results into (7) yields the reformulated Intermediate COCOMO model.

Four basic predesignated criteria were employed in order to demonstrate the effectiveness of the reformulation procedure, all of which are directed toward obtaining a 'close fit' between the data and the reformulated model:

(1) High coefficient of multiple determination R^2, in terms of the logarithmic model, (2), which provides a basis but not a guarantee that acceptable results will be obtained when transforming back to original units.

(2) Proximate values of the coefficient of determination, R'^2, between the results of the multiplicative model, (8), in original units and the published COCOMO estimates.

(3) Proximate values of the mean absolute percent error, MAPE, in original units between estimates obtained from (8) and the published COCOMO estimates.

(4) Proximate visual error pattern between the estimates obtained from (8) and the published COCOMO results.

In general, the coefficient of multiple determination, R^2, is defined as a measure of correlation with regard to linear relationships, and represents the proportion of the total sum of squares in the predicted variable explained or accounted for by the predictor, or explanatory, variables. Based upon all four criteria, the reformulation can be demonstrated to be effective, as is presented in detail in Gulezian [5].

The primary reason for including this section on existing models is to set the stage for potentially utilizing contributions from the existing state of the art into the new modeling process rather than employing a start from scrated approach at the outset. Due to the relative amounts of information available about each of the models, COCOMO contributes the most toward a beginning point, and actually can provide a structure which utilizes the essential features of the other models. This is the reason for introducing the concept of reformulation of COCOMO.

A large part of the problem stems from a dilemma that exists within the software cost estimating and modeling environment itself. On the one hand, cost estimates, and more objective estimates, are needed on a day to day basis. A modeled estimate presumeably provides more objectivity than, say, one by analogy or intuition. On the other hand, existing, publicly available models are not configured to a particular environment. The time and effort required to develop an environment-specific model is, to a certain extent, prohibitive mainly due to a lack of environment-specific data and the time required to plan for and collect it. Moreover, the existing state of the art in software modeling does not embrace the full complement of modeling tools available, and is such that a much needed incremental expansion capability is not built into the existing models nor the modeling process.

RECOMMENDED MODEL FEATURES

Mention was made earlier regarding some of the limitations of the models typically used to estimate software costs. In addition to overcoming the basic limitations, an effective model should possess numerous features not contained within models in use to date. Given below is a list of features that are considered in the approach presented here. It should be noted that there exists an implicit time frame associated with the items presented and that all features are not necessarily achievable at once, or simultaneously. The intent is to present modeling goals which are accompanied by a structure that allows for incorporation of the features incrementally when and if either the appropriate data is available or it is deemed essential to do so. The features considered are as follows:

(1) Embody the existing state of the art, to the extent possible, within the model. This is important for two reasons: (a) starting from scratch is impractial since some demonstrated capability and effort have been established and too much time initially is required, and (b) the existing models provide a reference frame for comparison.

(2) Use a starting point that provides estimates not necessarily employing new data but with a structure that can be built upon to develop an effective model. This is important in conjunction with (1) since a model with some of the features listed automatically is available in useable form and prior to the time consuming-data collection effort that is necessary to develop a more appropriate model.

(3) Utilize data based coefficient values that are simultaneously estimated in order to account for the interrelationships among all relevant variables, or parameters, and which are comparable to re-estimated values based upon subsequent model improvement efforts.

(4) The model should be re-configurable in order to accomodate new data or varying or changing environments or conditions.

(5) A defined measure of estimation error should be established for purposes of further model improvement and to meet the capability in (3).

(6) The capability to establish probabilistic interval estimates based upon properly developed measures of error should be available.

(7) The model should possess the capability to extrapolate and interpolate beyond the specific input values used to construct the model.

(8) The modeling process should possess the capability to add new cost drivers, metrics, and additional equations within the same structure.

(9) The equation form(s) should be modifiable and adaptable without altering the basic structure.

(10) The model should be based upon procedures that provide valid assessments of the importance and/or effectiveness of input variables or parameters in the estimation process.

(11) The model should accomodate both subjective and objective or measured inputs and metrics, depending upon availability.

(12) The model should accomodate for varying amounts and levels of inputs depending upon availability. As an example, this would include system levels, phases, and activities not initially considered.

(13) Capability of including any available sizing measures with or without lines of code as the principal cost driver should be incorporated within the model.

STATISTICAL MODELING

Assuming the proper preliminary problem defintion and design have been accomplished in conjunction with a requisite support environment, the fundamental steps that should be undertaken in any modeling process are as follows:

(1) Identification

(2) Estimation and Fitting

(3) Validation

(4) Application

(5) Iteration

Basically, identification refers to establishing the type of model to employ, which in the present context would refer to the functional equation form to be employed. This can be done either conceptually, data based, or as an eclectic combination of both. The estimation and fitting stage involves the establishment of specific numerical values for the coefficients, or multipliers so to speak, in the model that allows one to actually substitute into the model equation and obtain an estimate of the variable of interest, namely effort or cost in this case.

Validation of a model (as opposed to validation of data) relates to the determination that a model provides effective estimates within a particular estimating environment. Essentially validity refers to the practical value of the model under different circumstances. Three stages of the validation process exist: (1) during model development when a fitted model provides a yardstick that can reveal further aspects of the data used to develop the model, (2) during model testing when new data is gathered to provide further

validation of the model, and (3) during application, when introducing monitoring procedures to check whether the intial model remains satisfactory in use.

Application obviously refers to the use of a particular model. In this instance, the main goal is to provide cost or effort estimates. Further applications can be considered such as sensitivity analyses, "what-if" exercises, and various types of decision making.

The modeling process should be one of continuous development, since at any point in time a developed model is always tentative and is based upon the knowledge base used at that time. As new information from any source is obtained and with changes within the sofware development process itself, these should be fed back into the modeling process. This we refer to as iteration.

The steps in the modeling process delineated have been used in one combination or another with respect to any of the existing software cost estimation models. However, since a cost estimation model cannot be distinctly defined mathematically, or as error free, methods devised to understand the effects of error should be employed to develop more effective models, whether they be environment-specific or not. Such methods emanate from the discipline of statistics. It is the application of statistics in the modeling process that distinguishes the proposed approach from those currently applied to software cost estimation. This is not to say that statistical methods have never been applied to date, however, they have been applied sporadically and not to their fullest extent of usefulness.

Basically, full utilization of statistical methods can take the software cost estimation modeling process beyond its present status. In part, this can be accomplished by overcoming the limitations of the existing models discussed and by embodying the recommended features presented above. This is effected by methods that properly measure and account for the various types of error

associated with the estimation problem and the tests and comparisons that can be based upon these types of error.

Special statistical techniques and tests are available that apply to each of the general modeling steps itemized. Treatment of many of these methods can be found in Draper and Smith [4], Belsley, et. al. [2], and Morrison [7]. There is, however, no one particular technique that applies to each step, and in many cases choices have to be made regarding which techniques to employ. Consequently, it is not the intent here to delineate specific statistical methods that would be used in modeling cost. The remainder of this section and the next, concerning the interface among the existing state of the art, model features, and statistical methods, will address the types of analyses, comparisons, and tests that can be employed in order to accomplish the goal of developing more effective models.

As stated, the distinguishing feature that differentiates the use of statistics as opposed to other methods of modeling lies in the manner in which error is handled. In the case of software cost estimation, there are a number of types of error that must be considered. Of fundamental importance is the inherent error or variation in cost that naturally arises. At any point in time, or technological state, the magnitude of this error, although unknown, can be considered fixed for a particular environment. With respect to the cost estimation problem, there are in addition two types of prediction error, one associated with an estimate of the average cost of developing or maintaining a group of systems, projects, or components and that associated with estimating the cost of an individual entity. In addition, error potentially exists with respect to measurement of the input variables, or cost drivers, and the equation form used, or what is referred to as misspecification error.

When viewing the costs of a number of existing systems or components with respect to a given type of activity, we can think of the variation among these costs as total error. By modeling, regardless of the methods(s) used, one attempts to explain or account for this error. Presumeably, if this is explained it its entirety, one could estimate cost exactly. Since this is not possible due to the presence of inherent variability, the primary goal is to account for as much variation as possible in terms of relevant factors or drivers.

Although much more methodology is involved, the basic method proposed falls under the umbrella of multiple regression and correlation analysis, which deals with the relationships between a variable to be estimated or predicted and any number of explanatory or control variables. Overall, as a modeling process, it involves continually introducing variables and model features, testing, and re-analyzing data, all aimed at reducing the total error to the inherent error.

Regardless of the starting point, statistical methodology can make a contribution to each of the modeling stages or steps itemized above. Different contributions potentially would be made depending upon the starting point, which is to be discussed subsequently. Given below is a set of contributions or set of types of methods that can be used in the software cost estimation modeling process which go beyond the current state of the art and are directed toward more realistic and effective modeling. The list given is categorizd by modeling step, however, it should be recognized that many components of the list are not unique to a specific category. For example, many procedures falling under iteration actually are part of the validation process when moving from re-estimation, say, as part of model tailoring in order to determine whether a truly re-configured model has been developed.

The list of unique contributions and method types is as follows:

Identification

- Establish the appropriate equation form or structure based upon the data using specifically developed tests for this purpose. As new information is obtained, it is not necessary to adhere to the same equation form. Nor is it necessary to employ exactly the same form for different environments or systems or individual components.
- Establish a reference model that has a comparable format to later developed models so that an understanding or the potential reasons for departures can be established.
- Model identification primarily is data driven rather than imposed on a given set of data.
- The starting point for identifying the equation form is not necessarily fixed. In other words, one may begin with a simple linear form, an intuited equation form, or an existing model form and re-analyze the resulting data in order to add or delete terms or consider various transformations.

Fitting and Estimation

- Employ varying methods of fitting and estimation that potentially satisfy the needs of the problem at a particular stage or which are based upon different types of data available.
- Provide a consistent and automatic basis for calibrating a given form to other situations.

- Utilize methods that provide for analytically based measures of all types of error cited, as well as the error in estimating the individual coefficients or cost driver multipliers.
- Use error based procedures for determining the variables to be included in the model, or established variable selection procedures.
- The fitting and estimation procedures used are independent of variable scales, and can automatically incorporate data used for different models, or both subjective and measured or metric inputs.
- All coefficient estimates and re-estimates are simultaneously estimated based upon generalizeable procedures.
- Utilize established criteria and tests to determine the adequacy and degree of fit.
- Use established criteria or measures to determine the extent to which the cost drivers are interrelated.
- Employ appropriate sampling techniques to estimate newly introduced metrics, which may be independently estimated.

Validation

- Establish conditions under which a model is appropriate.
- Assess the potential importance of individual components and variables included in the model.
- Provide appropriate criteria and tests.
- Assess the practicality of a given model in terms of the amount of inherent error.
- Provide a probabilistic basis for performing sensitivity analyses relating of the estimated coefficients or cost driver multipliers.

- Establish the basis for combining model components or individual models and data bases.
- Provide a probabilistic basis for observed departures from a reconfigured model.

Application

- Establish estimates of average and individual effort or cost with appropriate measures of error, or corresponding probabilistic error bounds.
- Provide a probabilistic basis for project control.
- Provide the basis for extrapolating and interpolating with regard to variable inputs.
- The potential exists for developing a computer package that not only provides effort estimates but which is reconfigureable, based upon new data and changing dependencies within the estimating environment.

Interation

- Use established procedures which consistently can be used to tailor, augment, and expand existing models during various stages of development.
- Provide a probabilistic basis for comparing models at various stages of development or under different conditions in order to assess potential reasons for departure and establishing the form of a reconfigured model.

- Use probabilistic based methods of data exploration and diagnosis to assess outliers and influential observations or cases, or to assess model representation in a particular case.

Overall, the items delineated above are distinguished in terms of the use of a full body of consistently applied methods that yield comparable results generated through a process of estimation, re-estimation, and testing based upon probabilistically based procedures and criteria.

BASIC REQUIREMENTS

The extent to which the modeling process can be undertaken depends upon certain conditions relating to the data planned and collected. These conditions are dependent upon the following key items:

Observational unit

Number of observational units

Accounting additivity

Data item linkages

Data timing

Basic to the use of statistical modeling is the unit of observation from which data is observed or collected. Typically when relationships are observed among numerous variables or parameters, an element of similarity or commonality must exist among some standardized units of observation. In the case of the publicly available models, typically this unit is the software system. Frequently, due to their size, CSCI's within a system may be used consistently. Detailed COCOMO deals specifically with information at the

system, subsystem, and module levels however, in practice, the Intermediate form seems to be the one that is applied. Aside from definitional problems relating to the concept of a module, module level data typically has not been made available in useable form.

The key point of importance is that all single equation models to date use the same observational unit for the same equation. The level of detail acquired in such models is a function, to a certain extent, of the system level component employed. In the case of a development model, system level data is used, although efforts now are being undertaken to accumulate CSCI level data to be used in the publicly available models. The extent to which data is or can be made available within Sonar Systems depends upon whether the components undergo configuration control or not.

When dealing with maintenance data, there exists a problem not encountered in development. This relates to the fact that any fault repair or upgrade is implemented on a segment of code which is designated as a configuration item. This however, is not, in the sense used above, consistently a segment of code at a particular system level. Further, regardless of the component level actually affected by change, it is the configuration item that undergoes formal test and integration. Although the configuration item can be used as the unit of observation, additional identifiers or variables to account for the differences in configuration items ultimately should be included in the maintenance model equations for fault repairs and upgrades. This results from the fact that although all of the subsystems are structured according to the heirarchy of architectural elements delineated in DOD Standard 2167, different definitions of lower level components are used. Further, similar components do not uniformly exist consistently among all subsystems and in some cases lowest level components are not individually compileable. Consequently, in order to obtain

consistency across subsystems when considered within the same model structure, it is necessary to introduce variables or identifiers that will isolate the nature of the observational units employed and the relative magnitudes of the effort involved in each case.

Related to the problem of defining a common observational unit is the problem of the number of such units available. When developing a statistical rather than a heursitic model, the number of variables or parameters included in the model is bounded by the number of observational units. Actually, the larger the number of observational units relative to the number of variables the better, since the extra units are needed to measure the error or uncertainty associated with the model.

Two important points can be made that relate to the previous two paragraphs. First, in the case of models relating to maintenance effort or cost, any configuration item used in the maintenance activity can be considered as the observational unit, regardless of its definition within a particular subsystem. This runs contrary to the way in which models typically are developed at present. In the case of system or subsystem development, it is not necessary to consider this problem, although the second point does relate to this. In order to obtain a sufficient number of observational units it may be necessary to consider computer software configuration items (CSCI's) or top level computer software components (TLCSC's) as the observational units. This would also be the case if modeling efforts began with data regarding a single subsystem.

Presumeably, the more detail observed in the data, the better the model will be potentially. If the data is too aggregated, too little detail is available. This relates to the problems of additivity, linkage, and timing. If data is to be isolated by system level, phase, and WBS component it is necessary that effort or cost data be additive among the individual items

within the system, among phases, and organizational activities. Accompanying variable information must be attributable to the individual items as well. Moreover, individual components of system level, phase, and WBS must be linked to the observational unit to which they correspond. Consequently, the level of model detail is dependent upon the way data is available or the way in which it is planned at the outset.

Timing of the data collection effort is important for two reasons, especially if it is subjective. On the one hand, data collected at the end of a total effort tends to be more aggregated and less informative. On the other hand, obtaining data at the end of a total effort implies that one must wait until enough projects are completed before any actual environment specific reformulation or modeling can be accomplished. Preferably, data for each phase should be collected.

Another problem related to the timing of data collection is the nature of the data and of the modeling process. Although other uses exist, typically a model is to be used to provide an up-front or before the fact prediction, whereas data collected in order to build a model is collected after the fact. Generally, one seeks variables to include in the model whose values for a particular estimate are available up-front. In the case of software cost estimation there is a question as to the validity of after the fact input data since much of it is subjective or is in the form of an estimate. Consequently, before and after the fact input data should be collected in order to determine what the model actually is measuring. In other words, attempt to build the model prospectively on an incremental basis and then retrospectively make adjustments to the model to reflect differences between the two types of inputs.

The aternative ways in which modeled estimates of software costs can be obtained have been discussed in addition to the fact that these need not be mutually exclusive. Due to reasons already cited, it is beneficial to bring together parts of each alternative in order to meet any of the various Sonar Systems modeling needs. Overall, this amounts to using the reformulated COCOMO model described as a starting point, selecting the model need or needs toward which direction is to be focussed, delineating the data items needed to expand the initial model corresponding to the selected needs, and continually reconfiguring the initial model for each of the chosen needs.

By re-estimating the coefficients in (2) statistically, a reformulated reference model automatically embracing the features listed earlier is obtained, which also provides estimates comparable to COCOMO. Moreover, the same subjective inputs used in COCOMO can be used in the reformulated version, as well as objective or measured values which is not true of COCOMO. No new environment specific data base is required.

Since the COCOMO model designated in (1) primarily deals with the development phases and not operations and maintenance, it is implied that estimates using the initial reformulation in (2) are not directly applicable to maintenance initially; this, however, represents a feasible starting point. Consequently, the initial reformulation would provide estimates for all cases or needs at the outset. Moreover, it would posses all of the recommended features delineated earlier that are not possessed by any of the publicly available models.

Once specific data are collected that corresponds to a particular estimation need, separate equations or models with appropriate re-estimated numerical coefficients can be obtained for each need. At this point, the

separate need specific model forms can be applied and potentially validated. The entire process would continually be re-iterated as new and different types of information, additional subsystems, and added requirements are introduced in order to obtain newly reconfigured models.

Implied within the material presented is the fact that over time, individual models for each need with possibly different equation forms, different variables, and based upon specialized data sets would be developed. Each of these would undergo different forms of the iteration process which is suited to the individualized estimation need. The following presents the nature of the models that would correspond to each of Sonar Systems estimation needs without specifying the specific equation form and variables or data items ultimately required in each case. It should be emphasized that what is presented is merely a set of initial considerations. The lists provided neither give all of the types of variables that could be employed nor do individual items listed have to be employed. Furthermore, the time at which individual model components or variables would be introduced would vary and depend upon various factors not anticipateable at this time.

Annual Software Maintenance Budget

Assuming specific fault repairs and upgrades are anticipated for an ensuing year, the annual maintenance budget should equal the sum of the costs for all fault repairs and upgrades, or

$$\text{Annual Budgeted Mainteance Effort} = \Sigma \text{ Fault repair effort} + \Sigma \text{ Upgrade effort}$$

In such a case, it is then necessary to develop individual models to estimate the cost of individual fault repairs and the cost of individual upgrades. The

format of the separate models is provided in the appropriate sections below. Since there would exist some carryover of uncompleted fault repairs and upgrades from a previous year, either each such corresponding estimate must be modified in terms of an estimated complete figure or phase specific data must be isolated and linking variables must be introduced between the various phase activities and incorporated into separate phase effort model equations.

Five-Year Annual Moving Maintenance Budget

If the same assumption made above can be made for subsequent years, then a similar approach to that of the individual yearly budget can be used, and the individual annual estimates summed. A way of projecting percent complete figures and variations in capacity must be devised and applied to the subsequent years' estimates.

A more realistic approach would be to assume that the initial year could be estimated in terms of summing individual fault repairs and upgrades. In such a case an annual maintenance history must be developed before an appropriate model could be developed. In either case, if phase linked equations are used, a maintenance history must be accumulated.

Fault Repair and Upgrades

Although fault repairs and upgrades represent distinct problems with some overlap, they are considered together due to the more general nature of the material presented. Requisite details can be delineated if and when the modeling process is to be undertaken. Effort would be related to the following categories of variables:

Variables reflecting status and time system is in fleet

Fault history variables

Upgrade history variables

Fault type identifiers

Upgrade type identifiers

System maintenance history variables

Staffing/Capacity variables

Affected code and type variables

Special configuration item identifiers

A selection of development variables

Quality and maintainability metrics

Special integration and testing variables

Development and R & D Prototype

Although COCOMO is primarily a development cost model and has been introduced as the starting point for new model development, it should be re-emphasized that much room for improvement exists. On the one hand, the subjective ratings should eventually either be replaced or used in conjuction with measured varibles and/or metrics in order to obtain greater input accuracy. On the other hand, many more factors should be considered and tested in order to introduce more realism into the model. A preliminary list of variable categories used to expand and tailor the model is as follows;

Selection of COCOMO cost drivers

Selection of drivers and components from SLIM, Price-S, and System-3

Target productivity measures

Language

Computing environment and constraint variables

Contracting environment identifiers

Change history variables

Scheduling variables

System identifiers (functional, component, interface)

Documentation measures

Data type variables

Capacity/Staffing variables

Phase

WBS activity

System level

Compilation history

Metrics

Although similar activities would be related to the development of an R & D prototype, the fact that this is smaller and more controlled than a full-scale system or subsystem development would imply that a separate modeling effort should be carried. Tests can be made to compare the prototype and full development models in order to determine the extent to which similarities and differences exist between the two. Further, fewer observational units may be available in the case of a prototype which would place a restriction upon the candidate variables from which to choose.

Implementing an ECP During Development

Basically, implementing an engineering change proposal during full-scale development can be treated similar to implementing an upgrade during the

maintenance phase. More information must be acquired regarding the differences in procedure and staffing within Sonar Systems between the two phases. On the surface, it obviously would not be necessary to utilize fleet history data. Further, special scheduling and interfacing variables should be considered.

It should be noted that the reformulated model can immediately be augmented, starting with one single set of data items, to begin new modeling for the cost to implement an upgrade, development cost, R & D prototype development cost, and the cost to implement an ECP during development. Implied here is the fact that one can almost immediately begin to formally test the applicability of the initial model in these four cases, and start the process of tailoring and potentially reconfiguring the initial model to meet four of the seven needs at the outset. In all cases, model calibration and expansion require an entire environment specific data base.

The variable categories provided above for each of the cases have been provided in order to provide a partial guide regarding the potential magnitude of the modeling problem. As for as specific data items are concerned, a complete list of these for the four publicly available models can be found in Najberg [8]. Additional data items corresponding to the various categories listed are delineated in Thibodeau [9]. This also introduces potential quality and maintainability metrics to be considered.

SUPPORT REQUIREMENTS

Since the total effort required to undertake the modeling process for all of the estimating needs is a tremendous and time consuming task, certain choices must be made regarding what can be accomplished over time. Also, in order to undertake the modeling process in any case, a support environment is

necessary. The following represents a list of the requirements that must be considered:

(1) Selection of estimation needs to be addressed, initially and over time.

(2) Establish time frame over which modelling activity is or can be undertaken.

(3) Selection of the version of COCOMO to be used to obtain estimated outputs.

(4) Determination of sources and nature of input/output from other existing models regarding on-going comparative analyses and validation.

(5) Determination of level of model detail to be considered over time.

(6) Profile of configuration items currently in the system, either under development or maintenance.

(7) Profile fault repair history.

(8) Profile upgrade history.

(9) Determination of the extent to which Sonar Systems current DBMS can be used in the process.

(10) Status of plans for acquiring new DBMS and role in future modeling.

(11) Delineation of candidate data items to be considered and collected initially and over time.

(12) Determination regarding timing of data collection.

(13) Determination of the hardware configuration to be used for generating estimates, storing collected and generated information, and for data analysis.

(14) Determination regarding the use of canned packages for statistical analysis or a newly developed package interfaced with the models developed.

(15) Delineation of special purpose programs required and/or to be written to interface with the statistical package.

(16) Development of methods of data collection.

(17) Indentification of sources of various types of data.

(18) Determination of requirements for working DBMS and interface.

(19) Selection of subsystem(s) corresponding to which modeling is to apply.

(20) Determination of uniform categorization scheme to classify faults and upgrades.

Due to the long-term nature of the modeling process, the data base used to store the relevant information must accomodate data for projects long after their completion since the accumulated data over time would be re-used constantly in conjuction with newly acquired data. This would include newly introduced data items and metrics that are developed or uncovered as modeling progresses. This can be done in conjunction with the existing or planned automated data base for configuration management, or in a separate modeling data base. Since the data base must be interfaced with automated statistical procedures and must constantly be augmented, consideration should be given to the existence of a separate modeling or working data base, possibly as part of one developed through a canned computer package. This should be related to the nature of the computing facility used to perform the requisite analyses and store all intermediate modeling results and estimates.

Considerable attention must be given to the data collection effort required which supplements that automatically done for configuration control. An entire format must be developed considering whether additional data is to be collected in automated or manual form, the corresponding collection instruments, and other documents and personnel types supplying requisite data.

CONCLUDING REMARKS

The efforts underlying this report were directed toward developing a preliminary understanding and framework to be used as a basis to develop a cost estimating model for software maintenance and other software areas for the Sonar Systems Department. Due to the short duration of the effort, certain details and considerations by necessity have been omitted.

On the one hand, with repect to the annual maintenance budget estimates, it has been assumed that an ensuing year's faults and upgrades are known in advance. Based upon internal communications, this assumption appears to be reasonable and, therefore, the approach suggested directly applies. Whether the assumption is reasonable for an ensuing 5-year period remains questionable. Consequently, if one were to go beyond the usual treatment of applying some measure of change activity to aggregated budget figures in terms of a modeling effort, additional information must be obtained regarding internal policy and procedure and that related to capacity and productivity. In the case of the latter item, productivity measurement is considered to be a separate problem. On the other hand, without knowledge of specific fault repairs planned in advance, it is necessary to consider the development of an individual model to estimate the number of faults generated and entering maintenance.

Another consideration worth noting is that attention is anticipated with respect to the use of scheduling variables to estimate effort or cost in terms of future modeling efforts. No mention of a model to estimate schedule was considered since this was not stated internally to be of interest to date.

The philosophy underlying the modeling approach considered here, in addition to being unique in terms of its use of statistical methods throughout, is to begin with a simple model that is not dependent upon new

data which subsequently can be modified or extended incrementally based upon newly collected data and greater detail, when available or appropriate. Since no decisions were intended to be made by the principal investigator as a result of the current effort, the intent of the report is to provide the basis for making choices regarding the direction modeling will take within Sonar Systems.

In this regard, however, it is worth suggesting a reasonable starting point of a defined and doeable nature. Of the seven estimating needs, the quickest start would be in terms of modeling the cost of an upgrade. This results from the fact that the reformulated COCOMO model is immediately augmentable and data regarding upgrades is more readily available within the Department.

Coupled with the above is the possibility of developing an initial or starting model which is calibrated to a more appropriate data base. The most complete such data base which contains inputs for COCOMO and the other three models considered resides within the Electronic Surveillance Division at Hanscom Air Field [8]. Although it is based upon Air Force data rather than Naval sonar systems data, it potentially may be more appropriate than original COCOMO due to the size of the computer components and the fact that they deal with embedded systems. Actually, two reference models could be carried within the proposed structure to determine which is the more appropriate one to build upon.

Although all of the analytic capabilities of an existing packaged computer program cannot be realistically programmed, a simple self contained program can be developed which could include a reformulated version of COCOMO with similar estimating capabilities and one embodying the recommended features delineated earlier and requisite re-estimation algorithms to calibrate, augment and tailor initial or newly configured models.

Mention was made earlier in the report regarding the use of single equation models as opposed to ones with multiple equations. In many instances, different factors may affect different components of effort or cost, and as a consequence, it may be more realistic to use separate equations to describe the different relationships that potentially may exist.

Ideally, separate equations should be considered for each system level, phase, and WBS component combination, since the nature of the effort, and therefore possibly cost, varies within these combinations. This would apply not only to individual factors that may be relevant, but also to the functional form of the appropriate equations. Obviously, in some cases, an overlap may exist regarding individual factors across various equations. Moreover, individual equations may be linked, due to dependencies that exist among various phases and system levels, which also can be accomodated in the equation structure.

The ideas presented in the previous paragraph refer only to equations of effort or cost, which is to be estimated, as they relate to various relevant factors. Additional equations may be introduced that relate other relevant factors to be estimated to additional explanatory variables. Such estimating relationships may correspond to quality, documentation, schedule, and size. These in turn would be used to provide better estimates of these factors which would then be used in the effort equations to obtain further improvements. Ultimately, the entire set of equations can embrace the full software life cycle from requirements specification through operations and maintenance. Aside from introducing more realism throughout, the necessary linkage between maintenance and requirements and development would be effected.

Although the ideas suggested in the last paragraphs are desireable, implementation is not without its problems. On the one hand, the methods of estimation become much more involved and complex, Malinvaud [6] and Zellner

[10]. Furthermore, there exist numerous accounting problems that currently have not been addressed or resolved. This applies to proper and detailed effort allocations and becomes a greater problem when using measured inputs or metrics rather than subjective ratings.

REFERENCES

[1] Bailey, E.K. et.al., "A Descriptive Evaluation of Automated Software Cost-Estimation Models", IDA Paper p-1979, Contract Mda903-84-C-0031, Institute For Defense Analysis, October, 1986.

[2] Belsley, D.A., Kuh, E., Welsch, R.E., Regression Diagnostics, New York, NY: John, Wiley & Sons, Inc., 1980.

[3] Boehm, B.W., Software Engineering Economics, Englewood Cliffs, NJ: Prentice-Hall, 1981.

[4] Draper, N.R., Smith, H., Applied Regression Analysis, New York, NY: John Wiley and Sons, Inc., 1981.

[5] Gulezian, R.C., "Calibrating and Transporting" COCOMO, Submitted to IEEE Transactions on Software Engineering, February 1987.

[6] Malinvaud, E., Statistical Methods of Econometrics, Chicago: Rand McNally and Company, 1966.

[7] Morrison, D.F., Applied Linear Statistical Methods, Englewood Cliffs, NJ: Prentice-Hall, Inc., 1983.

[8] Najberg, A.C., "Software Data Base Development, Volume 1", The Analytic Sciences Corporation, Contract No. F33657-82-D0253/0014, June 1984.

[9] Thibodeau, R., Hughes, A.S. Software Quality Prediction and Assessment - Maintainability, General Research Corporation, Contract No. F30602-83-C-0026.

[10] Zellner, A., An Introduction to Bayesian Inference in Econometrics, New York: John Wiley and Sons, Inc., 1971.

[11] System-3 Featuring the JS-3: Software Estimation and Schedule Estimating Plus Decision Support Users Manual. Computer Economics Inc., June 1986.

[12] Price S-3 Reference Manual (Revision II), October 1984.

[13] SLIM User Manual for the IBM PC, July 1984.

[14] AN/SQQ-890/V Surface ASN Combat System Life Cycle Plan, February 13, 1987.

[15] AN/SQQ-89 Test Authorization, Software Support, CPCI 03610100, September 15, 1987.